ICS 35.040
L 80

中华人民共和国国家标准

GB/T 17903.2—2008/ISO/IEC 13888-2:1998
代替 GB/T 17903.2—1999

信息技术　安全技术　抗抵赖
第2部分:采用对称技术的机制

Information technology—Security techniques—Non-repudiation—Part 2:Mechanisms using symmetric techniques

(ISO/IEC 13888-2:1998,IDT)

2008-06-26 发布　　2008-11-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会　发布

中 华 人 民 共 和 国
国 家 标 准
信息技术 安全技术 抗抵赖
第 2 部分:采用对称技术的机制
GB/T 17903.2—2008/ISO/IEC 13888-2:1998
*
中国标准出版社出版发行
北京复兴门外三里河北街 16 号
邮政编码:100045
网址 www.spc.net.cn
电话:68523946 68517548
中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销
*
开本 880×1230 1/16 印张 1.25 字数 30 千字
2008 年 9 月第一版 2008 年 9 月第一次印刷
*
书号:155066·1-33734 定价 18.00 元

GB/T 17903.2-2008

前　言

GB/T 17903 在总标题《信息技术　安全技术　抗抵赖》下，由以下几部分组成：

——第 1 部分：概述；

——第 2 部分：采用对称技术的机制；

——第 3 部分：采用非对称技术的机制。

本部分是 GB/T 17903 的第 2 部分，等同采用 ISO/IEC 13888-2:1998《信息技术　安全技术　抗抵赖　第 2 部分：采用对称技术的机制》，仅有编辑性修改。ISO/IEC 13888-2:1998 是由联合技术委员会 ISO/IEC JTC 1(信息技术)分技术委员会 SC 27(IT 安全技术)提出的。

本部分代替 GB/T 17903.2—1999《信息技术　安全技术　抗抵赖　第 2 部分：采用对称技术的机制》。本部分与 GB/T 17903.2—1999 相比，主要差异如下：

——本部分根据第 1 部分的修订，更改部分术语。

——本部分对部分叙述进行了文字修订，并修正了第 10 章中的“NORT”。

本部分的附录 A 是资料性附录。

本部分由全国信息安全标准化技术委员会提出并归口。

本部分主要起草人：中国科学院软件研究所、信息安全国家重点实验室。

本部分主要起草人：张振峰、冯登国。

本部分所代替标准的历次版本发布情况为：

——GB/T 17903.2—1999。

目　次

信息技术　安全技术　抗抵赖
第2部分:采用对称技术的机制

1　范围

抗抵赖服务旨在生成、收集、维护、利用和验证有关已声称事件或动作的证据,以解决有关该事件或动作已发生或未发生的争议。本部分描述了用于抗抵赖服务的通用结构,以及一些特定的、与通信有关的机制,用于提供原发抗抵赖(NRO)、交付抗抵赖(NRD)、提交抗抵赖(NRS)和传输抗抵赖(NRT)等。其他抗抵赖服务可用第8章描述的通用结构来构建,以满足安全策略的要求。

本部分依赖于可信第三方来防止欺诈性的抵赖。一般需要在线的可信第三方。

抗抵赖机制提供的协议用于交换各种抗抵赖服务规定的抗抵赖权标。本部分中使用的抗抵赖权标由安全信封和附加数据组成。抗抵赖权标作为抗抵赖信息予以存储,以备之后发生争议时使用。

依据特定应用的有效抗抵赖策略以及该应用操作所处的法律环境,抗抵赖信息可能包括以下附加信息:

a)　包括时间戳机构提供的可信时间戳在内的证据;

b)　公证人提供的证据,以确保动作或事件是由一个或多个实体执行或参与的。

抗抵赖只能在特定应用及其法律环境下、有明确定义的安全策略的范围内才可生效。

2　规范性引用文件

下列文件中的条款通过GB/T 17903的本部分的引用而成为本部分的条款。凡是注明日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本部分,然而,鼓励根据本部分达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本部分。

GB 15852—1995　信息技术　安全技术　用块密码算法作密码校验函数的数据完整性机制(idt ISO/IEC 9797:1994)

GB/T 15843.4—1999　信息技术　安全技术　实体鉴别　第4部分:采用密码校验函数的机制(idt ISO/IEC 9798-4:1997)

GB/T 18238.1—2000　信息技术　安全技术　散列函数　第1部分:概述(idt ISO/IEC 10118-1:1994)

GB/T 17903.1—2008　信息技术　安全技术　抗抵赖　第1部分:概述(ISO/IEC 13888-1:2004,IDT)

3　术语和定义

GB/T 17903.1—2008中的术语和定义适用于本部分。

4　记法和缩略语

4.1　记法

4.1.1　GB/T 17903.1—2008中的记法

$Imp(y)$	数据串y的印迹,或者是数据串y的散列码,或者是数据串y
$SENV_x$	使用实体X的秘密密钥x生成的安全信封

text	可构成一部分权标的数据项，包括密钥标识符和(或)消息标识符等附加信息
$y \| z$	y 和 z 按顺序的连接

4.1.2 本部分专用的记法

a	仅为实体 A 和可信第三方(TTP)所知的密钥
A	实体 A 的可区分标识符
b	仅为实体 B 和 TTP 所知的密钥
B	实体 B 的可区分标识符
d_a	交付机构(DA)的密钥
f, f_i	标明抗抵赖服务类型的数据项(标记)
$MAC_X(y)$	使用实体 X 的密钥对数据 y 计算而得到的密码校验值
m	待生成证据的消息
Pol	适用于证据的抗抵赖策略的可区分标识符
T_i	事件或动作发生的日期和时间
T_g	证据生成的日期和时间
ttp	仅为 TTP 所知的密钥，用于生成抗抵赖权标
x	为两个实体共知或者仅为 TTP 所知的密钥
z_1	由提供 NRO 权标的有关数据字段组成的数据字段
z_2	由提供 NRD 权标的有关数据字段组成的数据字段
z_3	由提供 NRS 权标的有关数据字段组成的数据字段
z_4	由提供 NRT 权标的有关数据字段组成的数据字段
z_5	由提供 TST 权标的有关数据字段组成的数据字段

4.2 缩略语

DA	Delivery Authority 交付机构
GNRT	Generic Non-Repudiation Token 通用抗抵赖权标
NRD	Non-Repudiation of Delivery 交付抗抵赖
NRDT	Non-Repudiation of Delivery Token 交付抗抵赖权标
NRO	Non-Repudiation of Origin 原发抗抵赖
NROT	Non-Repudiation of Origin Token 原发抗抵赖权标
NRS	Non-Repudiation of Submission 提交抗抵赖
NRST	Non-Repudiation of Submission Token 提交抗抵赖权标
NRT	Non-Repudiation of Transport 传输抗抵赖
NRTT	Non-Repudiation of Transport Token 传输抗抵赖权标
PON	Positive Or Negative 肯定或否定，验证过程的结果
TSA	Time-Stamping Authority 时间戳机构
TST	Time-Stamping Token 时间戳权标
TTP	Trusted Third Party 可信第三方

5 要求

5.1 如果两个实体使用本部分规定的某个机制，双方必须信任同一个第三方。

5.2 在使用这些机制之前，假定每个实体与可信第三方共享一个密钥。此外，可信第三方持有一个仅为自己所知的密钥。

注：密钥管理、密钥生成和密钥建立机制在 ISO/IEC 11770 中规定。

5.3 抗抵赖服务中的所有实体共享一个公共函数 *Imp*。函数 *Imp* 或者是恒等函数，或者是

GB/T 18238.1—2000 中定义的抗碰撞散列函数。

5.4 为创建安全信封而选取的 *MAC* 函数必须为抗抵赖服务的所有参与者所持有。

5.5 生成抗抵赖权标的 TTP 必须能够访问时间和日期。

5.6 本部分规定的机制的强度依赖于密钥的长度和保密性、依赖于函数 *MAC* 的性质以及校验值的长度。这些参数的选取应该满足安全策略规定的安全级别的需求。

6 本部分各章的组织

本部分描述的机制要求每一个相关实体都可以与 TTP 单独进行通信。他们需要使用第 7 章描述的安全信封。关于生成和验证抗抵赖权标的基本概念，以及证据的概念，在第 8 章描述。第 9 章描述的机制需要使用 TTP，每一个证据的生成与验证都需要 TTP 的参与。该机制的三种变型在第 10 章中作为例子进一步描述。

7 安全信封

共享一个秘密密钥的两个实体(该密钥仅为这两个实体所知)可以使用一种称为安全信封(SENV)的数据完整性校验方法来相互传递消息。SENV 使用秘密密钥来产生，用于保护输入数据项。SENV 也可由 TTP 使用仅为 TTP 持有的秘密密钥来生成和验证证据。

下面使用对称的完整性技术创建安全信封。实体 X 的秘密密钥 x 用于计算密码校验值 $MAC_X(y)$，该值附加在数据 y 的后面：

$SENV_X(y) = y \parallel MAC_X(y)$，

其中 $MAC_X(y)$可以是 GB 15852—1995 中规定的消息认证码。

函数 *MAC* 应满足 GB/T 15843.4—1999 中规定的下列要求：

——对任意密钥 x 和数据串 y，计算 $MAC_X(y)$是可行的；

——对任意固定的密钥 x，在预先不知道 x 的情况下，即使已知一组满足 $MAC_X(y_i)=z_i(i=1,2,\cdots)$的$(y_i,z_i)$(其中 y_i 值可以在得到 $z_j(j=1,2,\cdots,i-1)$之后进行选择)，找到一对新的(y',z)使得 $MAC_X(y')=z$ 是计算上不可行的。

8 抗抵赖权标的生成和验证

在本章描述的抗抵赖机制中，TTP 担当证据生成和证据验证机构的角色。它可信赖地维护某些记录的完整性并直接参与解决争议。

8.1 TTP 创建权标

TTP 颁发与消息 m 相应的“权标”。权标是一种安全信封，由 TTP 使用其秘密密钥作用于该消息所确定的数据而形成。因为其他实体都不知道秘密密钥 ttp，TTP 是唯一可以创建或者验证权标的实体。在 GB/T 17903.1—2008 中，通用抗抵赖权标(GRNT)定义如下：

$\mathrm{GRNT} = text \parallel SENV_X(y)$。

此外，在发布权标之前，TTP 应该检查证据请求中的数据项。

8.2 抗抵赖机制使用的数据项

8.2.1 安全信封使用的数据项

安全信封

$SENV_X(z) = z \parallel MAC_X(z)$

将在本部分描述的抗抵赖机制中进行交换，下列数据字段构成了该安全信封的内容：

$z = Pol \parallel f_i \parallel A \parallel B \parallel C \parallel D \parallel E \parallel T_g \parallel T_i \parallel Q \parallel Imp(m)$。

数据字段 z 包括以下数据项：

Pol　　适用于证据的抗抵赖策略的可区分标识符

f_i	提供的抗抵赖服务的类型
A	原发实体的可区分标识符
B	与原发实体进行交互的实体的可区分标识符
C	证据生成者的可区分标识符
D	证据请求者的可区分标识符，如果证据请求者与原发实体不同
E	动作涉及到的其他实体的可区分标识符
T_g	证据生成的日期和时间
T_i	事件或动作发生的日期和时间
Q	需要保护的可选数据
$Imp(m)$	与动作有关的消息 m 的印迹(可能是消息 m 的散列码，也可能是消息 m 本身)。

注：根据抗抵赖策略的不同，有些数据项是可选的。

8.2.2 抗抵赖权标使用的数据项

抗抵赖权标包括一个文本域，记为 $text$：

抗抵赖权标 $= text \parallel SENV_{TTP}(z)$

$text$ 包括一些不需要密码保护但在计算完整性校验值 MAC 时要用到的、标识消息和密钥的附加数据(如消息标识符或密钥标识符)。本信息依赖于所使用的技术。

8.3 抗抵赖权标

证据由抗抵赖权标提供，如果策略要求，由附加权标提供，如时间戳权标(TST)、或者由另一个可信的第四方(如公证人)提供的、对事件和动作以及消息的存在性给予附加保证的权标。

如果可信第三方可以独自生成可信时间戳，则不需要增加时间戳权标(TST)作为证据。抗抵赖权标(NROT、NRDT、NRST 和 NRTT)中包含的时间可认为是安全可靠的，因为它是由可信机构提供的。

如果可信第三方(TTP、DA)不能够提供可信时间戳，那么抗抵赖信息集合中就需要增加由可信时间戳机构(TSA)提供的时间戳权标(TST)以完成证据。

8.3.1 原发抗抵赖权标

原发抗抵赖权标(NROT)由 TTP 应原发者的请求而创建。

NROT $= text \parallel z_1 \parallel MAC_{TTP}(z_1)$

其中

$z_1 = Pol \parallel f_1 \parallel A \parallel B \parallel C \parallel D \parallel T_g \parallel Q \parallel Imp(m)$。

NROT 所需信息 z_1 包括如下数据项：

Pol	适用于证据的抗抵赖策略的可区分标识符
f_1	原发抗抵赖的标记
A	原发者的可区分标识符
B	预定接收者的可区分标识符
C	生成证据的 TTP 的可区分标识符
D	观察者的可区分标识符，如果存在独立观察者
T_g	证据生成的日期和时间
Q	需要保护的附加数据
$Imp(m)$	消息 m 的印迹

8.3.2 交付抗抵赖权标

交付抗抵赖权标(NRDT)由 TTP 应接收者的请求而创建。

NRDT $= text \parallel z_2 \parallel MAC_{TTP}(z_2)$

其中

$z_2 = Pol \parallel f_2 \parallel A \parallel B \parallel C \parallel D \parallel T_g \parallel T_2 \parallel Q \parallel Imp(m)$。

NRDT 所需信息 z_2 包括如下数据项：

Pol	适用于证据的抗抵赖策略的可区分标识符
f_2	交付抗抵赖的标记
A	原发者的可区分标识符
B	接收者的可区分标识符
C	证据生成者的可区分标识符
D	观察者的可区分标识符，如果存在独立观察者
T_g	证据生成的日期和时间
T_2	消息交付的日期和时间
Q	需要保护的附加数据
$Imp(m)$	消息 m 的印迹

8.3.3 提交抗抵赖权标

提交抗抵赖权标(NRST)由交付机构(DA)创建。交付机构是一个可信第三方，可以与生成 NROT 或 NRDT 的是同一个机构。

$NRST = text \parallel z_3 \parallel MAC_{DA}(z_3)$

其中

$z_3 = Pol \parallel f_3 \parallel A \parallel B \parallel C \parallel D \parallel E \parallel T_g \parallel T_3 \parallel Q \parallel Imp(m)$。

NRST 所需信息 z_3 包括如下数据项：

Pol	适用于证据的抗抵赖策略的可区分标识符
f_3	提交抗抵赖标记
A	原发者(提交实体)的可区分标识符
B	预定接收者的可区分标识符
C	交付机构(DA)的可区分标识符
D	观察者的可区分标识符，如果存在独立观察者
E	代表交付机构进行活动的机构的可区分标识符(可选的)
T_g	证据生成的日期和时间
T_3	消息提交的日期和时间
Q	需要保护的附加数据
$Imp(m)$	提交待传输的消息 m 的印迹

8.3.4 传输抗抵赖权标

传输抗抵赖权标(NRTT)由交付机构 DA 生成。

$NRTT = text \parallel z_4 \parallel MAC_{DA}(z_4)$

其中

$z_4 = Pol \parallel f_4 \parallel A \parallel B \parallel C \parallel D \parallel E \parallel T_g \parallel T_4 \parallel Q \parallel Imp(m)$。

NRTT 所需信息 z_4 包括如下数据项：

Pol	适用于证据的抗抵赖策略的可区分标识符
f_4	传输抗抵赖的标记
A	原发者的可区分标识符
B	接收者的可区分标识符
C	交付机构的可区分标识符
D	观察者的可区分标识符，如果存在独立观察者
E	代表交付机构进行活动的机构的可区分标识符(可选的)
T_g	证据生成的日期和时间

T_4	消息交付到接收者的数据存储区的日期和时间
Q	需要保护的附加数据
$Imp(m)$	消息 m 的印迹

8.3.5 时间戳权标

时间戳权标(TST)由时间戳机构创建,其定义如下:

$TST = text \parallel z_5 \parallel MAC_{TSA}(z_5)$

其中

$z_5 = Pol \parallel f_5 \parallel TSA \parallel T_g \parallel T_5 \parallel Q \parallel Imp(m)$。

数据字段 z_5 包括如下数据项:

Pol	适用于证据的抗抵赖策略的可区分标识符
f_5	时间戳权标的标记
TSA	时间戳机构的可区分标识符
T_g	为特定消息生成证据(如 TST)的日期和时间
Q	需要保护的附加数据
$Imp(m)$	与时间戳相关的消息 m 的印迹

8.4 TTP 进行的权标验证

在抗抵赖交换过程的某个环节上,可能需要 TTP 对实体的权标(如上述定义所示)进行验证。在交换完成以后的某个时刻,也可能需要再次验证权标,或者向第四方提供证据以证明其真实性。

验证过程不仅要检验权标是否由 TTP 创建,而且要检验权标是否与消息的数据字段确切相关。为了验证权标是否为给定的消息而创建,实体把由消息计算而得的 $Imp(m)$ 与数据字段 z 中包括的 $Imp(m)$ 进行比较,然后要求 TTP 对权标及其数据字段进行验证。

为了验证由对称完整性技术生成的安全信封,进行如下操作:使用实体 X 的相应秘密密钥 x 对安全信封中包含的数据 y 重新计算密码校验值 $MAC_X(y)$,然后把结果与所提供的密码校验值进行比较。

TTP 提供了两种验证权标的方法。

8.4.1 在线权标验证

本方法中,TTP 使用包含秘密密钥 ttp 的安全模块来验证权标。安全模块将该权标与使用数据项 z_i 和秘密密钥 ttp 在其内部生成的值进行比较,并返回比较结果,该结果决定了权标是否有效。由于密钥 ttp 不为 TTP 之外的任何人所知,如果安全模块返回的结果表明该权标是有效的,那么所验证的权标也可以认为是真实可信的。

8.4.2 权标表

本方法中,TTP 发布的所有权标存储在一张表中。对每个已创建的权标,TTP 记录下权标和相关的数据字段(z_i)以及秘密密钥 ttp 的密钥标识符。要验证一个权标,TTP 把该权标作为索引在表中查找。如果在表中能够找到要验证的权标,而且该权标所带的数据字段(即权标的一部分)与表中对应的数据字段相符,则认为该权标是真实有效的。

9 特定抗抵赖机制

本章的抗抵赖机制支持生成下列抗抵赖证据:原发抗抵赖(NRO)、交付抗抵赖(NRD)、提交抗抵赖(NRS)和传输抗抵赖(NRT)。另外,本章定义了时间戳的生成机制。实体 A 想要向实体 B 发送消息,于是 A 就成为抗抵赖传输的原发者,实体 B 成为接收者。

本章所描述的某些机制中,请求的 z_i 数据字段不包含时间信息。时间信息由 TTP(或 DA)提供,或者由时间戳机构应 TTP(或 DA)的请求而提供。

注:当 $Imp(m)$ 与消息 m 相同时,不必将 m 与权标一起发送,并且验证 $Imp(m)$ 的步骤也可省略。

9.1 原发抗抵赖机制

原发者创建了一条消息并发送给特定的接收者。接收者使用 TTP 来验证与之相关的原发抗抵赖权标，从而检验该消息来源于其声称的发送者。

本机制的第一步，原发者构造数据并封装入 SENV 发送给 TTP。TTP 生成原发抗抵赖权标(NROT)并返回给 A。第二步，原发者 A 把 NROT 与消息 m 发送给接收者 B。第三步，接收者把安全信封中封装的 NROT 发给 TTP 进行验证。原发抗抵赖在第三步建立。

9.1.1 步骤 1：在原发者 A 和 TTP 之间

a) 实体 A 使用密钥 a 生成安全信封 $SENV_A(z'_1)$，其中 z'_1 同 8.3.1 规定的 z_1，但数据项 T_g 为空。实体 A 把安全信封发送给 TTP 以请求 NROT；

b) TTP 验证安全信封来自实体 A。如果验证通过，TTP 插入数据项 T_g 以完成 z_1，并使用密钥 ttp 计算：

$\mathrm{NROT} = text \parallel z_1 \parallel MAC_{TTP}(z_1)$，

然后将 $SENV_A$(NROT)返回给 A；

c) 实体 A 验证 $SENV_A$(NROT)来自 TTP。

9.1.2 步骤 2：从原发者 A 到接收者 B

原发者 A 向实体 B 发送：$m \parallel$ NROT。

9.1.3 步骤 3：接收者 B 与 TTP 之间

a) 实体 B 验证 z_1 中的 $Imp(m)$值，然后使用密钥 b 生成 $SENV_B$(NROT)并发送给 TTP，要求验证来自 A 的 NROT；

b) TTP 验证 $SENV_B$(NROT)来自 B，并验证 NROT 是合法的。如果 $SENV_B$(NROT)无效，机制终止。如果 $SENV_B$(NROT)是有效的，TTP 向 B 发送 $SENV_B$(PON $\parallel$ NROT)，其中：如果 NROT 是合法的，PON 为肯定，如果 NROT 不合法，则 PON 为否定；

c) 实体 B 检验 $SENV_B$(PON $\parallel$ NROT)来自 TTP；若检验通过，并且验证结果 PON 为肯定，则建立了原发抗抵赖；

d) 存储 NROT 以供将来原发抗抵赖使用。

9.2 交付抗抵赖机制

本机制的第一步，实体 B 在接收到消息 m 后，向 TTP 发送请求以要求生成交付抗抵赖权标，该请求封装在安全信封中。TTP 生成交付抗抵赖权标(NRDT)，并返回给接收者 B。第二步，接收者 B 发送 NRDT 给原发者 A。第三步，原发者将 NRDT 封装在安全信封中发送给 TTP 进行验证。交付抗抵赖在第三步建立。

9.2.1 步骤 1：在接收者 B 与 TTP 之间

a) 实体 B 使用密钥 b 生成安全信封 $SENV_B(z'_2)$，其中 z'_2 同 8.3.2 规定的 z_2，但 T_g 为空。实体 B 通过发送安全信封向 TTP 请求 NRDT；

b) TTP 检验安全信封是否来自实体 B。如果是，TTP 插入数据项 T_g 以完成 z_2，并利用密钥 ttp 计算：

$\mathrm{NRDT} = text \parallel z_2 \parallel MAC_{TTP}(z_2)$；

c) 实体 B 验证 $SENV_B$(NRDT)来自 TTP。

9.2.2 步骤 2：接收者 B 和原发者 A 之间

实体 B 向实体 A 发送：NRDT。

9.2.3 步骤 3：在原发者 A 和 TTP 之间

a) 实体 A 验证 z_2 中的 $Imp(m)$值，然后使用密钥 a 生成 $SENV_A$(NRDT)并发送给 TTP，要求验证来自 B 的 NRDT；

b) TTP 验证 $SENV_A$(NRDT)来自 A，并验证 NRDT 的合法性。如果 $SENV_A$(NRDT)无效，机

制终止。如果 $SENV_A$(NRDT)是有效的,TTP 向 A 发送 $SENV_A$(PON ‖ NRDT),其中:如果 NRDT 是合法的,PON 为肯定;若 NRDT 不合法,PON 为否定;

c) 实体 A 检验 $SENV_A$(PON ‖ NRDT)是否来自 TTP。如果是,并且验证结果 PON 是肯定的,则交付抗抵赖建立;

d) 存储 NRDT 以供将来的交付抗抵赖使用。

9.3 提交抗抵赖机制

本机制的第一步,提交实体 X 向交付机构 DA 发送消息 m 要求向前递送。第二步,交付机构 DA 发送提交抗抵赖权标(NRST)给实体 X。提交抗抵赖在第二步建立。

9.3.1 步骤 1:从提交实体 X 到交付机构 DA

a) 实体 X 使用密钥 x 生成安全信封 $SENV_X(z'_3)$,其中 z'_3 同 8.3.3 规定的 z_3,但 T_g 为空。实体 X 然后把安全信封和消息 m 一起发送给 DA 以请求 NRST;

b) DA 验证安全信封是否来自实体 X,并通过检查 $Imp(m)$来验证消息 m 的合法性,如果两者都是合法的,DA 插入数据项 T_g 以完成 z_3,并利用密钥 d_a 计算:

$$NRST = text \parallel z_3 \parallel MAC_{DA}(z_3)。$$

9.3.2 步骤 2:从交付机构 DA 到实体 X

a) DA 把 $SENV_X$(NRST)返回给提交实体 X;

b) 实体 X 验证 $SENV_X$(NRST)是来自 DA 的;

c) 如果验证通过,则存储 NRST 作为提交抗抵赖的证据(表明消息已经提交)。

9.4 传输抗抵赖机制

本机制的第一步,发送实体 X 向交付机构 DA 发送消息 m,要求向前递送。第二步,交付机构 DA 把消息 m 发送给接收实体 Y。第三步,交付机构生成传输抗抵赖权标(NRTT),并发送给消息 m 的原发实体 X。传输抗抵赖在第三步建立。

9.4.1 步骤 1:从实体 X 到交付机构

实体 X 向交付机构 DA 发送消息 m,并请求 NRTT 权标。

9.4.2 步骤 2:从交付机构到实体 Y

交付机构 DA 发送消息 m 给实体 Y。

9.4.3 步骤 3:从交付机构到实体 X

a) 交付机构 DA 利用密钥 d_a 生成 NRTT:

$$NRTT = text \parallel z_4 \parallel MAC_{DA}(z_4),$$

其中 z_4 见 8.3.4 的规定;

b) 交付机构 DA 向实体 X 发送 $SENV_X$(NRTT);

c) 实体 X 检验 $SENV_X$(NRTT)及其内容。如果是有效的,则存储 NRTT 作为传输抗抵赖的证据(即消息已经交付给预定的接收者 Y)。

9.5 获取时间戳的机制

时间戳权标(TST)由一个可信的时间戳机构(TSA)应实体 X 的要求而生成。

第一步,请求实体 X 发送消息 z'_5,由 TSA 插入时间 T_g 使其完整。第二步,TSA 将时间戳权标(TST)返回给请求实体。

9.5.1 步骤 1:从实体 X 到时间戳机构 TSA

a) 实体 X 使用密钥 x 生成安全信封 $SENV_X(z'_5)$,其中 z'_5 同 8.3.5 规定的 z_5,但 T_g 为空。然后实体 X 向 TSA 发送安全信封以请求时间戳权标;

b) TSA 生成包含日期和时间的 T_g,把数据项 z'_5 完成为 z_5;

c) TSA 生成 TST:

$$TST = text \parallel z_5 \parallel MAC_{TSA}(z_5)。$$

9.5.2 步骤 2:从时间戳机构 TSA 到实体 X

a) 时间戳机构利用实体 A 和 TSA 共知的密钥 x 生成安全信封 $SENV_X$(TST),由此把时间戳权标返回给请求实体 X;

b) 实体 X 验证安全信封的有效性。

10 抗抵赖机制实例

本章所示的抗抵赖机制可在实体 A 和 B 之间提供原发抗抵赖和交付抗抵赖。实体 A 欲向实体 B 发送消息,于是成为抗抵赖交换的原发者。作为消息的接收方,实体 B 就是接收者。在使用本机制之前,假设实体 A 和实体 B 分别持有密钥 a 和 b,TTP 除了拥有自己的密钥 ttp 以外,还持有密钥 a 和 b。

下面给出了使用在线 TTP 的三种不同的抗抵赖机制(M1,M2 和 M3)。

注 1:通过在 $SENV$ 消息中包含时间戳或者序列号,可以防止未授权延迟或消息重放。通过在 NROT 和 NRDT 中包含时间戳,可进一步验证消息传输时的时间戳。

注 2:如果 $Imp(m)$ 与消息 m 相同,则不必把 m 与权标一起发送,并且可省略验证 $Imp(m)$ 的步骤。

10.1 机制 M1:强制 NRO,可选 NRD

在两个实体与 TTP 之间通过 3 个步骤可建立原发抗抵赖,如果继续可选的 NRD 步骤(根据接收者的决定),那么再进行 2 个步骤可建立交付抗抵赖(见图 1)。

注:尽管是否继续进行交付抗抵赖的步骤取决于接收者,但要注意的是,一旦建立了交付抗抵赖,这一可选的交付抗抵赖就完全绑定了。

图 1 机制 M1

10.1.1 步骤 1:原发者 A 与 TTP 之间

a) 实体 A 使用密钥 a 生成安全信封 $SENV_A(z'_1)$,其中 z'_1 同 8.3.1 规定的 z_1,但数据项 T_g 为空。然后实体 A 把安全信封发送给 TTP 以请求 NROT;

b) TTP 检查安全信封是否来自 A。如果是,TTP 插入数据项 T_g 以完成 z_1,并使用密钥 ttp

计算：

NROT = $text \parallel z_1 \parallel MAC_{TTP}(z_1)$，

然后将 $SENV_A$(NROT)返回给 A；

c) 实体 A 验证 $SENV_A$(NROT)来自 TTP。

10.1.2 步骤 2：从原发者 A 到接收者 B

原发者 A 向实体 B 发送：$m \parallel$ NROT。

10.1.3 步骤 3：接收者 B 与 TTP 之间

a) 实体 B 验证 z_1 中包含的 $Imp(m)$值，然后使用密钥 b 生成 $SENV_B$(NROT)并发送给 TTP，要求验证来自 A 的 NROT；

b) TTP 检查 $SENV_B$(NROT)和 NROT。如果两者都是有效的，TTP 生成交付抗抵赖权标 NRDT，并发送 $SENV_B$(PON ‖ NROT ‖ NRDT)给实体 B，其中 PON 是肯定的。如果 $SENV_B$(NROT)是有效的，而 NROT 无效，TTP 发送 $SENV_B$(PON ‖ NROT)给实体 B，其中 PON 是否定的；

c) 实体 B 验证 $SENV_B$(PON ‖ NROT ‖ NRDT)来自 TTP，如果是有效的，而且 PON 是肯定的，则创建了原发抗抵赖(即消息来自于 A)。如果 B 接收到的是 $SENV_B$(PON ‖ NROT)并且 PON 是否定的，那么 NROT 是无效的，机制终止；

d) 存储 NROT 以供将来原发抗抵赖使用。

10.1.4 步骤 4：从接收者 B 到原发者 A

实体 B 发送 NRDT 给实体 A。

10.1.5 步骤 5：在原发者 A 与 TTP 之间

a) 实体 A 验证 z_2 中包含的 $Imp(m)$值，然后使用密钥 a 生成 $SENV_A$(NRDT)并发送给 TTP，要求验证来自 B 的 NRDT；

b) TTP 验证 $SENV_A$(NRDT)是否来自 A，并验证 NRDT 的真实性。如果两者都是有效的，TTP 发送 $SENV_A$(PON ‖ NRDT)给 A，其中 PON 为肯定的。如果 NRDT 是无效的，TTP 向 A 发送 $SENV_A$(PON ‖ NRDT)，其中 PON 是否定的；

c) 实体 A 验证 $SENV_A$(PON ‖ NRDT)是否来自 TTP。如果有效，并且验证值 PON 是肯定的，则建立了交付抗抵赖；

d) 存储 NRDT 以供将来的交付抗抵赖使用。

10.2 机制 M2：强制 NRO，强制 NRD

在两个实体和 TTP 之间通过 4 个步骤可建立原发抗抵赖和交付抗抵赖。在本机制中，TTP 在向 B 发送消息收据的同时，直接通过 $SENV$ 把它发送给 A(见图 2)。

10.2.1 步骤 1：原发者 A 和 TTP 之间

a) 实体 A 使用密钥 a 生成安全信封 $SENV_A(z'_1)$，其中 z'_1 同 8.3.1 规定的 z_1，但数据项 T_g 为空。实体 A 然后向 TTP 发送安全信封以请求 NROT；

b) TTP 验证安全信封是否来自 A。如果是，TTP 插入数据项 T_g 以完成 z_1，然后使用密钥 ttp 计算：

NROT = $text \parallel z_1 \parallel MAC_{TTP}(z_1)$，

进而利用密钥 a 生成 $SENV_A$(NROT)并返回给 A；

c) 实体 A 验证 $SENV_A$(NROT)来自 TTP。

10.2.2 步骤 2：从原发者 A 到接收者 B

实体 A 向实体 B 发送：$m \parallel$ NROT。

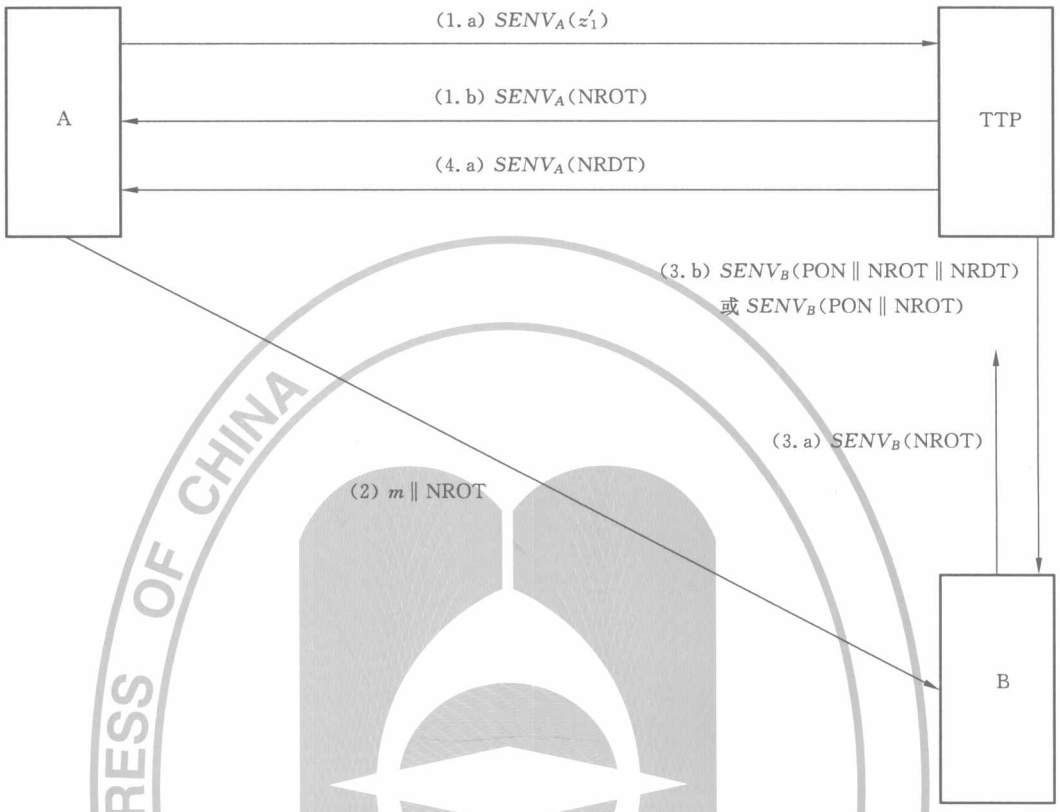

图 2 机制 M2

10.2.3 步骤 3:接收者 B 与 TTP 之间

a) 实体 B 验证 z_1 中包含的 $Imp(m)$值,然后使用密钥 b 生成 $SENV_B$(NROT)并发送给 TTP,要求验证来自 A 的 NROT;

b) TTP 检验 $SENV_B$(NROT)是否来自 B,并检验 NROT 是否真实的。如果两者都有效,TTP 生成 NRDT,并发送 $SENV_B$(PON ‖ NROT ‖ NRDT)给实体 B,其中 PON 是肯定的;如果 $SENV$ 是有效的,而 NROT 无效,TTP 发送 $SENV_B$(PON ‖ NROT)给 B,其中 PON 是否定的;

c) 实体 B 验证 $SENV_B$(PON ‖ NROT ‖ NRDT)是否来自 TTP;如果是,而且 PON 是肯定的,则创建了原发抗抵赖。相应的,如果 B 接收到了 $SENV_B$(PON ‖ NROT),而且 PON 是否定的,那么 NROT 无效,机制终止;

d) 存储 NROT 以供将来原发抗抵赖使用。

10.2.4 步骤 4:TTP 和实体 A 之间

a) 在步骤 3 中向 B 发送 NRDT 后,TTP 立即向 A 发送 $SENV_A$(NRDT);

b) 实体 A 检查 $SENV_A$(NRDT)和 NRDT,如果两者都有效,则建立了交付抗抵赖(即 B 收到了消息);

c) 存储 NRDT 以供将来交付抗抵赖使用。

10.3 机制 M3:带有中介 TTP 的强制 NRO 和 NRD

在两个实体和 TTP 之间通过 4 个步骤可建立原发抗抵赖和交付抗抵赖。在机制 M3 中,TTP 在原发者和接收者之间充当了中间人的角色,两个实体不再直接通信。为此,实体 A 发送消息给 TTP 作为步骤 1 中的一部分,TTP 将之传递给实体 B 作为步骤 2 的一部分(见图 3)。

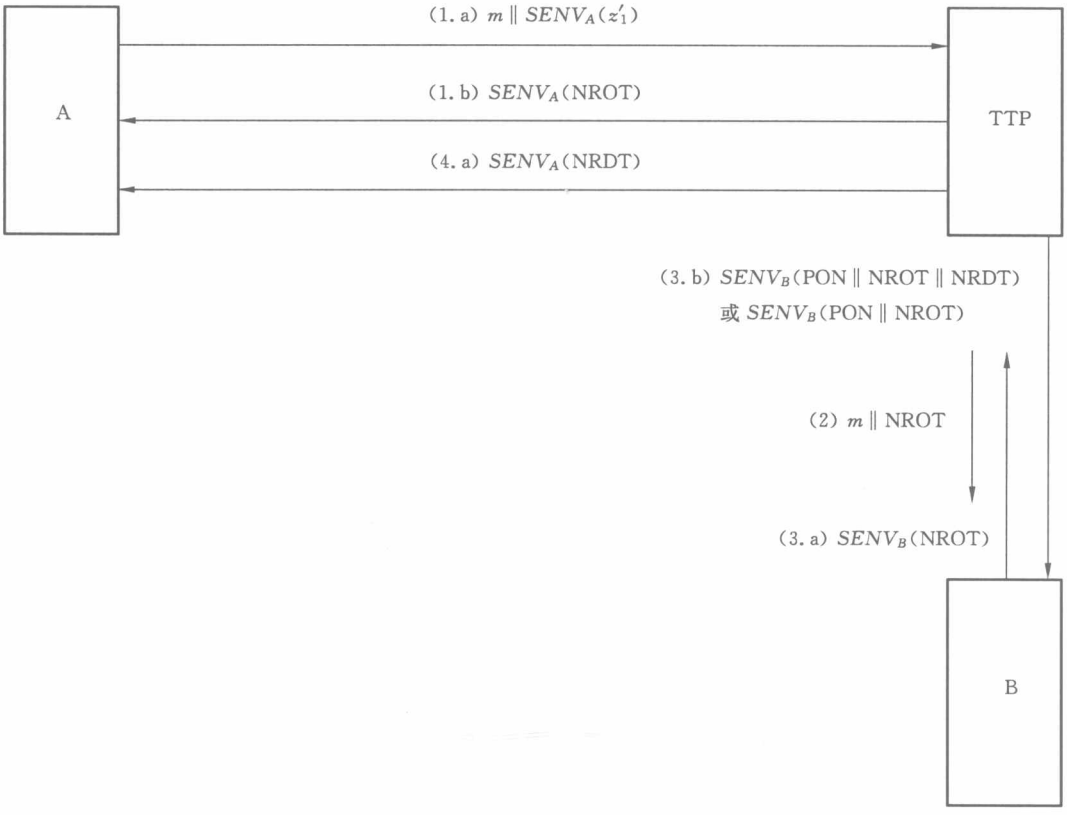

图 3　机制 M3

10.3.1　步骤 1:原发者 A 和 TTP 之间

a)　实体 A 使用密钥 a 生成安全信封 $SENV_A(z'_1)$,其中 z'_1 同 8.3.1 规定的 z_1,但数据项 T_g 为空。实体 A 把安全信封和消息 m 一起发送给 TTP 以请求 NROT;

b)　TTP 检验安全信封是否来自 A。如果是,TTP 插入数据项 T_g 以完成 z_1,并使用密钥 ttp 计算:

$\text{NROT} = text \| z_1 \| MAC_{TTP}(z_1)$,

之后利用密钥 a 将 $SENV_A$(NROT)返回给 A;

c)　实体 A 验证 $SENV_A$(NROT)来自 TTP。

10.3.2　步骤 2:从 TTP 到接收者 B

TTP 将 m 和 NROT 发送给 B。

10.3.3　步骤 3:实体 B 与 TTP 之间

a)　由于 NROT 不是以安全信封的方式收到的,实体 B 必须与 TTP 一起来验证 NROT,所以 B 在验证 $Imp(m)$之后,向 TTP 发送 $SENV_B$(NROT);

b)　TTP 验证 $SENV_B$(NROT)来自 B,并验证 NROT 的真实性。如果两者都是有效的,TTP 创建 NRDT,并向 B 发送 $SENV_B$(PON ‖ NROT ‖ NRDT),其中 PON 为肯定。如果 $SENV$ 是有效的,而 NROT 无效,TTP 向 B 发送 $SENV_B$(PON ‖ NROT),其中 PON 是否定的;

c)　实体 B 检验 $SENV$ 是否来自 TTP。如果是,而且 NROT 是肯定的,则创建了原发抗抵赖。相应地,如果 B 接收到的是 $SENV_B$(PON ‖ NROT),其中 PON 是否定的,那么 NROT 无效,机制终止;

d)　存储 NROT 以供将来原发抗抵赖使用。

10.3.4 步骤4:TTP和原发者A之间

a) 在步骤3中向B发送NRDT后,TTP立即向A发送$SENV_A$(NRDT);

b) 实体A在验证$SENV_A$(NRDT)确实来自TTP后,建立交付抗抵赖;

c) 存储NRDT以供将来交付抗抵赖使用。